Fonctionnement de l'internationalisation

Résumé :

Ce document est dédié à l'internationalisation de *Code_Aster*, c'est-à-dire l'ensemble des outils, les pratiques, l'organisation qui permet au code de gérer des messages et des affichages dans différentes langues ainsi que la traduction de la documentation.

La condition principale pour obtenir une traduction de qualité des messages (quand bien même elle n'est pas automatique mais humaine) est d'avoir des messages originaux compréhensibles, bien écrits en français. C'est pour cette raison que l'on détaille également la vérification faite sur les messages originaux.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

1 Traduction des messages

1.1 Fonctionnement

La prise en charge des messages en plusieurs langues s'appuie sur les outils standards de `gettext`.
On donne ici les grandes lignes du fonctionnement de gettext. On se reportera à la documentation de ce produit pour plus de précisions.

Les messages d'erreurs sont rassemblés dans des catalogues (confer [D4.06.01]).
Le module `i18n.py` (du package `Execution`) est en charge du choix de la langue d'affichage et de « l'installation » de la fonction de conversion des messages (il s'agit de la fonction « _ »).

La langue choisie par défaut est fonction de l'environnement d'exécution : variables d'environnement `LANG`, `LANGUAGE`, etc. (se référer à la documentation de `gettext` pour l'ordre de priorité et les différentes variables). La même variable est utilisée pour déterminer la manière d'afficher les caractères non-ascii, accentués en particulier. Il s'agit de l'encodage `utf-8` ou `iso-8859-1`).
L'utilisateur peut forcer le choix de la langue dans la commande `DEBUT` (confer [U4.11.01]).
C'est ce module qui définit le *domaine* `gettext` et donc le nom du fichier dans lequel les messages traduits seront cherchés. On utilise `aster_` + le nom de la version, soit par exemple : `aster_stable` pour la version d'exploitation, `aster_testing` pour la version stabilisée de développement.

Les outils de `gettext` (`pygettext` en l'occurrence) permettent d'extraire les messages des fichiers sources du code afin de produire un fichier POT, par exemple : `aster_unstable.pot`.
Pour chaque langue, on crée une copie de ce fichier complété des traductions des messages, nommés fichiers PO, par exemple : `aster_unstable.en.po`.
Ces fichiers PO sont « compilés » afin de produire les fichiers dits MO qui contiennent les messages, leurs traductions et une table de hashage pour y accéder. Ce sont ces fichiers MO qui sont utilisés lors d'une exécution de *Code_Aster*. Ils sont installés dans `$ASTER_ROOT/share/locale/`, dans un sous-répertoire dédié à chaque langue, soit par exemple : `en/LC_MESSAGES/aster_unstable.mo`.
À des fins de tests essentiellement, on peut choisir un autre répertoire en définissant la variable d'environnement `ASTER_LOCALEDIR` au lieu de `$ASTER_ROOT/share/locale/`.

En cours d'exécution, quand on doit afficher un message, la fonction « _ » est appelée avec en argument la chaine du message original (celui est qui écrit dans le code source, dans le catalogue de messages en général). Si ce message est présent **à l'identique** dans le fichier MO, alors la fonction « _ » retourne le message tel qu'il a été traduit (dans le mécanisme d'impression des messages, ce texte est ensuite formaté, complété de variables le cas échéant avant d'être affiché).
Si le fichier MO n'est pas trouvé ou bien si le message n'est pas trouvé, c'est le message original qui est affiché quelque soit la langue choisie (donc en français).
Si un message est modifié dans le code source, même juste un caractère, il est nécessaire que les fichiers PO et MO soient regénérés sinon la traduction ne sera pas affichée.

1.2 Organisation

Les fichiers PO ne peuvent pas être gérés en phase avec le code source. Il y a forcément un décalage entre la restitution du message en français et la mise à jour de la traduction (du moins tant que l'on n'exige pas aux développeurs de traduire simultanément leurs messages dans toutes les langues supportées).
D'ailleurs dans les projets libres, ce sont souvent des équipes différentes qui gèrent les traductions.

Un dépôt de source a été créé à l'adresse https://bitbucket.org/code_aster/codeaster-i18n.
Il permet de diffuser les dernières versions disponibles des fichiers POT et PO (pour l'anglais dans un premier temps).

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Aujourd'hui, le dépôt est géré par le responsable de cette fonctionnalité. Il pourra demain être transféré à une équipe externe qui voudra s'y investir. Les contributeurs peuvent *cloner* ce dépôt, puis proposer à l'administrateur du dépôt principal de récupérer et d'inclure leurs contributions (*pull request*).

L'objectif est de proposer des traductions aussi complètes que possible pour les versions `stable` et `testing`, afin de les fournir et les installer dans les paquets de *Code_Aster*, à chaque stabilisation.
La traduction de la version `unstable` est par nature continuellement en chantier du fait de la grande volatilité des messages chaque semaine.

2 Vérification des messages

La vérification des messages est effectuée par un cas-test dédié à cela : `supv002`.
Le but est de s'assurer :

- que les messages fournis dans les catalogues ont le bon format : ils doivent être en *unicode* pour être capable de gérer les caractères non-ascii ;
- qu'ils pourront être formatés correctement, notamment que les variables qu'ils contiennent ont un type cohérent ;
- que les messages sont écrits en français !

Pour ce dernier point, on procède à une vérification systématique du texte des messages à l'aide du vérificateur d'orthographe `aspell`. Celui-ci s'appuie sur son propre dictionnaire auquel on ajoute le jargon propre à Code_Aster. C'est-à-dire :

- un dictionnaire contenant la liste de tous les mots-clés du catalogue de commandes,
- un dictionnaire ajoutant un certain nombre de mots en complément du dictionnaire français de base (des noms propres comme *Newton*, des mots anglais mais adoptés dans notre jargon comme *level-set*, etc.).

On veillera à limiter au strict minimum le dictionnaire personnel pour *Code_Aster*.

Remarque

> *`aspell` ne prend en compte qu'un dictionnaire personnel. Lors de la vérification, on concatène automatiquement nos deux dictionnaires pour n'en former qu'un seul.*

Les dictionnaires sont rangés dans `$ASTER_ROOT/share/codeaster` respectivement sous les noms `code_aster_cata.aspell.per` et `code_aster_dict.aspell.per`. Attention à conserver l'encodage correct de ces fichiers.

2.1 Dictionnaire des mots-clés du catalogue

On construit ce dictionnaire de la manière la plus simple qui soit (on peut le mettre à jour de temps en temps en fonction de l'apparition de nouveaux mots-clés) : on prend tous les mots-clés en majuscules de `cata.py`.

`aspell` n'accepte pas les « _ » en milieu de mot, ni les mots contenant les chiffres. On peut produire le dictionnaire ainsi :

```
import os
import re
cata=open('cata.py', 'r').read()
exp = re.compile('([A-Z][A-Z]+)', re.M)
kw = list(set(exp.findall(cata)))
kw.sort()
kw.insert(0, 'personal_ws-1.1 fr 0 utf-8')
open('dict.aspell.per', 'w').write(os.linesep.join(kw))
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3 Traduction de la documentation

La traduction de la documentation est faite de manière automatique via le logiciel Systran.

Une licence perpétuelle de Systran Entreprise Server a été acquise. Le logiciel est installé sur le serveur hébergeant différents services de la forge logicielle (claut681.der.edf.fr). Il s'agit de la version 7.3. Une version 7.4 est attendue, elle devrait améliorer le rendu de la traduction et corriger quelques anomalies (ne gênant pas la compréhension de la version anglaise).

Des scripts, sauvegardés dans une base Mercurial, assurent la traduction de tous les documents chaque nuit. Ils sont associés aux scripts permettant la publication des documents. Les différentes étapes sont les suivantes (encapsulée dans *docaster_master.py*):

- Extraction de tous les documents français au format OpenOffice ;
- Traduction des documents : décompression des odt, envoi du fichier content.xml au serveur, recompression des odt avec la traduction (*systran_translator.py*) ;
- Conversion des documents français et anglais au format PDF ;
- Vérification des clés et des titres ;
- Mise en ligne sur le site www.code-aster.org.

Ces étapes sont assurées pour la version de développement (*unstable*) et la version d'exploitation (*stable*).

La documentation en anglais est accessible depuis l'onglet Documentation en utilisant le switch de langue. Un avertissement est donné concernant le non engagement sur le résultat de la traduction (*Warning : The translation process used on this website is a "Machine Translation". It may be imprecise and inaccurate in whole or in part and is provided as a convenience*). Cet avertissement figure sur la page d'accès aux documents et est systématiquement repris sur chaque pied de page.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)